## AMENDMENTS TO THE CLAIMS

For the convenience of the Examiner, all claims have been presented whether or not an amendment has been made. The claims have been amended as follows:

1. **(Previously Presented)** A method of detecting a computer virus, comprising:

emulating computer executable code in a subject file;

detecting at least one modification to a memory state of a computer system, wherein the at least one modification:

is caused by the emulation of the computer executable code; and

comprises installation of an exception handler or an interrupt handler.

2. **(Previously Presented)** The method of Claim 1, wherein:

the at least one modification comprises installation of an exception handler; and

the emulated computer executable code comprises instructions for forcing a corresponding exception.

3. **(Previously Presented)** The method of Claim 1, further comprising:

detecting writing of a pointer to at least one predetermined address in a system memory for storing an exception handler pointer.

4. **(Previously Presented)** The method of Claim 1, further comprising:

detecting installation, in a system memory, of a pointer to an exception handler.

5. **(Previously Presented)** The method of Claim 1, wherein:

the at least one modification comprises installation of an interrupt handler; and

the emulated computer executable code comprises instructions for forcing a corresponding interrupt.

6. **(Previously Presented)** The method of Claim 1, further comprising:

detecting writing of a pointer to at least one predetermined address in a system memory for storing an interrupt handler pointer.

7. **(Previously Presented)** The method of Claim 1, further comprising:

detecting use of a predetermined instruction to retrieve an address in a system memory corresponding to an interrupt descriptor table.

8. **(Previously Presented)** A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method for detecting a computer virus, the method comprising:

emulating computer executable code in a subject file;

detecting at least one modification to a memory state of a computer system, wherein the at least one modification:

is caused by the emulation of the computer executable code; and

comprises installation of an exception handler or an interrupt handler.

9. **(Previously Presented)** A computer system, comprising:

a processor; and

a program storage device readable by a computer system, tangibly embodying a program of instructions executable by the processor to perform a method for detecting a computer virus, the method comprising:

emulating computer executable code in a subject file;

detecting at least one modification to a memory state of a computer system, wherein the at least one modification:

is caused by the emulation of the computer executable code; and

comprises installation of an exception handler or an interrupt handler.

10. **(Previously Presented)** A computer data signal embodied in a transmission medium which embodies a program of instructions executable by a computer for detecting a computer virus, comprising:

a first segment comprising emulation code to emulate computer executable code in a subject file; and

a second segment comprising detector code to detect at least one modification to a memory state of a computer system, wherein the at least one modification:

is caused by the emulation of the computer executable code; and

comprises installation of an exception handler or an interrupt.

11. **(Previously Presented)** An apparatus for detecting computer viruses, comprising:

an emulator component operable to emulate computer executable code in a subject file; and

a detector component operable to detect at least one modification to a memory state of a computer system, wherein the at least one modification:

is caused by emulation of the computer executable code; and

comprises installation of an exception handler or an interrupt handler.

12. **(Previously Presented)** The apparatus of Claim 11, wherein the detector component is further operable to monitor a system memory.

13. **(Previously Presented)** The apparatus of Claim 11, wherein the at least one modification comprises installation of an exception handler.

14. **(Previously Presented)** The apparatus of Claim 13, wherein the emulated computer executable code comprises instructions forcing a corresponding exception.

15. **(Previously Presented)** The apparatus of Claim 11, wherein the at least one modification comprises writing of a pointer to at least one predetermined address in a system memory for storing an exception handler pointer.

16. **(Previously Presented)** The apparatus of Claim 11, wherein the at least one modification comprises installation of an interrupt handler.

17. **(Previously Presented)** The apparatus of Claim 16, wherein the emulated computer executable code comprises instructions for forcing a corresponding interrupt.

18. **(Previously Presented)** The apparatus of Claim 11, wherein the at least one modification comprises writing of a pointer to at least one predetermined address in a system memory for storing an interrupt handler pointer.

19. **(Previously Presented)** The apparatus of Claim 11, wherein the at least one modification comprises use of a predetermined instruction to retrieve an address in a system memory corresponding to an interrupt descriptor table.

20. **(Previously Presented)** The method of Claim 1, wherein the computer system comprises a first memory component and a second memory component, and wherein access to the second memory component is more restricted than access to the first memory component.

21. **(Previously Presented)** The method of Claim 20, wherein the exception handler or the interrupt handler attempts to modify the second memory component.